

WE CLAIM:

1. A computer program product for triggering an operation at a destination computer using data transferred between a source computer and said destination computer, said computer program product comprising:

receiving code operable to receive at said destination computer operation specifying XML data sent by said source computer;

parsing code operable to parse said operation specifying XML data to identify one or more complex data types within said operation specifying XML data;

matching code operable to match the or each complex data type with an associated execution process available to said destination computer; and

triggering code operable to trigger processing by the or each execution process associated with a complex data type within said operation specifying XML data.

2. A computer program product as claimed in claim 1, wherein parameter data used by an execution process is represented by data within said complex data type of said execution process.

3. A computer program product as claimed in claim 1, wherein said operation performed includes making a call to an API available to said destination computer.

4. A computer program product as claimed in claim 1, wherein said operation performed includes configuring said destination computer to execute a computer program.

5. A computer program product as claimed in claim 4, wherein said execution process is operable to map configuration data specified within said operation specifying XML data to a configuration data store of said destination computer.

6. A computer program product as claimed in claim 5, wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

a database entry.

7. A computer program product as claimed in claim 1, wherein an identifier of an execution process within said complex data type includes at least one of:

- 5 data specifying a computer file operable to trigger said execution process;
 data specifying a communication channel operable to trigger said execution process; and
 data specifying an operating system command operable to trigger said execution process.

10 8. A computer program product as claimed in claim 1, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process.

15 9. A computer program product as claimed in claim 8, wherein said result data includes data specifying existing configuration data of said destination computer.

20 10. A computer program product as claimed in claim 9, wherein said execution process is operable to map existing configuration data of said destination computer stored within a configuration data store of said destination computer to said result data to be returned to said source computer.

25 11. A computer program product as claimed in claim 10, wherein said configuration data store is one of:

- a Windows Registry entry;
 an INI file;
 a DAPI store; and
 a database entry.

30 12. A computer program product as claimed in claim 10, wherein said result data is passed from said destination computer to said source computer as XML data.

13. A computer program product as claimed in claim 1, wherein said operation includes returning result data from said destination computer to said source computer

in dependence upon whether or not said execution process is available to said destination computer.

14. A computer program product as claimed in claim 1, wherein an operation that may be performed by said destination computer includes installing a new execution process.

15. A computer program product as claimed in claim 1, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

16. A computer program product for triggering an operation at a destination computer using data transferred between a source computer and said destination computer, said computer program product comprising:

data forming code operable to form at said source computer operation specifying XML data containing one or more complex data types;
transmitting code operable to transmit from said source computer to said destination computer said operation specifying XML data; wherein
the or each complex data type within said operation specifying XML data corresponds to an execution process available to said destination computer to be triggered to operate.

17. A computer program product as claimed in claim 16, wherein parameter data used by an execution process is represented by data within said complex data type of said execution process.

18. A computer program product as claimed in claim 16, wherein said operation performed includes making a call to an API available to said destination computer.

19. A computer program product as claimed in claim 16, wherein said operation performed includes configuring said destination computer to execute a computer program.

20. A computer program product as claimed in claim 19, wherein said execution process is operable to map configuration data specified within said operation specifying XML data to a configuration data store of said destination computer.

5 21. A computer program product as claimed in claim 20, wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

10 a database entry.

22. A computer program product as claimed in claim 16, wherein an identifier of an execution process within said complex data type includes at least one of:

data specifying a computer file operable to trigger said execution process;

15 data specifying a communication channel operable to trigger said execution process; and

data specifying an operating system command operable to trigger said execution process.

20 23. A computer program product as claimed in claim 16, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process.

24. A computer program product as claimed in claim 23, wherein said result data
25 includes data specifying existing configuration data of said destination computer.

25. A computer program product as claimed in claim 24, wherein said execution process is operable to map existing configuration data of said destination computer stored within a configuration data store of said destination computer to said result data
30 to be returned to said source computer.

26. A computer program product as claimed in claim 25, wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;
a DAPI store; and
a database entry.

27. A computer program product as claimed in claim 25, wherein said result data is passed from said destination computer to said source computer as XML data.

28. A computer program product as claimed in claim 16, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

29. A computer program product as claimed in claim 16, wherein an operation that may be performed by said destination computer includes installing a new execution process.

30. A computer program product as claimed in claim 16, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

31. A method of triggering an operation at a destination computer using data transferred between a source computer and said destination computer, said method comprising the steps of:

receiving at said destination computer operation specifying XML data sent by said source computer;

parsing said operation specifying XML data to identify one or more complex data types within said operation specifying XML data;

matching the or each complex data type with an associated execution process available to said destination computer; and

triggering processing by the or each execution process associated with a complex data type within said operation specifying XML data.

32. A method as claimed in claim 31, wherein parameter data used by an execution process is represented by data within said complex data type of said execution process.

33. A method as claimed in claim 31, wherein said operation performed includes making a call to an API available to said destination computer.

34. A method as claimed in claim 31, wherein said operation performed includes configuring said destination computer to execute a computer program.

35. A method as claimed in claim 34, wherein said execution process is operable to map configuration data specified within said operation specifying XML data to a configuration data store of said destination computer.

36. A method as claimed in claim 35, wherein said configuration data store is one of:

- a Windows Registry entry;
- an INI file;
- a DAPI store; and
- a database entry.

37. A method as claimed in claim 31, wherein an identifier of an execution process within said complex data type includes at least one of:

- data specifying a computer file operable to trigger said execution process;
- data specifying a communication channel operable to trigger said execution process; and
- data specifying an operating system command operable to trigger said execution process.

38. A method as claimed in claim 31, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process.

39. A method as claimed in claim 38, wherein said result data includes data specifying existing configuration data of said destination computer.

40. A method as claimed in claim 39, wherein said execution process is operable to map existing configuration data of said destination computer stored within a configuration data store of said destination computer to said result data to be returned to said source computer.

41. A method as claimed in claim 40, wherein said configuration data store is one of:

- a Windows Registry entry;
- an INI file;
- a DAPI store; and
- a database entry.

42. A method as claimed in claim 40, wherein said result data is passed from said destination computer to said source computer as XML data.

43. A method as claimed in claim 41, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

44. A method as claimed in claim 41, wherein an operation that may be performed by said destination computer includes installing a new execution process.

45. A method as claimed in claim 41, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

46. A method of triggering an operation at a destination computer using data transferred between a source computer and said destination computer, said method comprising the steps of:

forming at said source computer operation specifying XML data containing one or more complex data types;

transmitting from said source computer to said destination computer said operation specifying XML data; wherein

the or each complex data type within said operation specifying XML data corresponds to an execution process available to said destination computer to be triggered to operate.

47. A method as claimed in claim 46, wherein parameter data used by an execution process is represented by data within said complex data type of said execution process.

48. A method as claimed in claim 46, wherein said operation performed includes making a call to an API available to said destination computer.

49. A method as claimed in claim 46, wherein said operation performed includes configuring said destination computer to execute a computer program.

50. A method as claimed in claim 49, wherein said execution process is operable to map configuration data specified within said operation specifying XML data to a configuration data store of said destination computer.

51. A method as claimed in claim 50, wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

a database entry.

52. A method as claimed in claim 46, wherein an identifier of an execution process within said complex data type includes at least one of:

data specifying a computer file operable to trigger said execution process;

data specifying a communication channel operable to trigger said execution process; and

data specifying an operating system command operable to trigger said execution process.

53. A method as claimed in claim 46, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process.

54. A method as claimed in claim 53, wherein said result data includes data specifying existing configuration data of said destination computer.

55. A method as claimed in claim 54, wherein said execution process is operable to map existing configuration data of said destination computer stored within a configuration data store of said destination computer to said result data to be returned to said source computer.

56. A method as claimed in claim 55, wherein said configuration data store is one of:

- a Windows Registry entry;
- an INI file;
- a DAPI store; and
- a database entry.

57. A method as claimed in claim 55, wherein said result data is passed from said destination computer to said source computer as XML data.

58. A method as claimed in claim 46, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

59. A method as claimed in claim 46, wherein an operation that may be performed by said destination computer includes installing a new execution process.

60. A method as claimed in claim 46, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

61. Apparatus for triggering an operation at a destination computer using data transferred between a source computer and said destination computer, said apparatus comprising:

receiving logic operable to receive at said destination computer operation
 5 specifying XML data sent by said source computer;
 parsing logic operable to parse said operation specifying XML data to identify one or more complex data types within said operation specifying XML data;
 matching logic operable to match the or each complex data type with an associated execution process available to said destination computer; and
 10 triggering logic operable to trigger processing by the or each execution process associated with a complex data type within said operation specifying XML data.

62. Apparatus as claimed in claim 61, wherein parameter data used by an execution process is represented by data within said complex data type of said
 15 execution process.

63. Apparatus as claimed in claim 61, wherein said operation performed includes making a call to an API available to said destination computer.

20 64. Apparatus as claimed in claim 61, wherein said operation performed includes configuring said destination computer to execute a computer program.

65. Apparatus as claimed in claim 64, wherein said execution process is operable to map configuration data specified within said operation specifying XML data to a
 25 configuration data store of said destination computer.

66. Apparatus as claimed in claim 65, wherein said configuration data store is one of:

a Windows Registry entry;
 30 an INI file;
 a DAPI store; and
 a database entry.

67. Apparatus as claimed in claim 61, wherein an identifier of an execution process within said complex data type includes at least one of:

data specifying a computer file operable to trigger said execution process;

data specifying a communication channel operable to trigger said execution process; and

data specifying an operating system command operable to trigger said execution process.

68. Apparatus as claimed in claim 61, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process.

69. Apparatus as claimed in claim 68, wherein said result data includes data specifying existing configuration data of said destination computer.

70. Apparatus as claimed in claim 69, wherein said execution process is operable to map existing configuration data of said destination computer stored within a configuration data store of said destination computer to said result data to be returned to said source computer.

71. Apparatus as claimed in claim 70, wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

a database entry.

72. Apparatus as claimed in claim 70, wherein said result data is passed from said destination computer to said source computer as XML data.

73. Apparatus as claimed in claim 61, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

74. Apparatus as claimed in claim 61, wherein an operation that may be performed by said destination computer includes installing a new execution process.

75. Apparatus as claimed in claim 61, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

76. Apparatus for triggering an operation at a destination computer using data transferred between a source computer and said destination computer, said apparatus comprising:

data forming logic operable to form at said source computer operation specifying XML data containing one or more complex data types;

transmitting logic operable to transmit from said source computer to said destination computer said operation specifying XML data; wherein

the or each complex data type within said operation specifying XML data corresponds to an execution process available to said destination computer to be triggered to operate.

77. Apparatus as claimed in claim 76, wherein parameter data used by an execution process is represented by data within said complex data type of said execution process.

78. Apparatus as claimed in claim 76, wherein said operation performed includes making a call to an API available to said destination computer.

79. Apparatus as claimed in claim 76, wherein said operation performed includes configuring said destination computer to execute a computer program.

80. Apparatus as claimed in claim 79, wherein said execution process is operable to map configuration data specified within said operation specifying XML data to a configuration data store of said destination computer.

81. Apparatus as claimed in claim 80, wherein said configuration data store is one of:

- a Windows Registry entry;
- an INI file;
- a DAPI store; and
- a database entry.

82. Apparatus as claimed in claim 76, wherein an identifier of an execution process within said complex data type includes at least one of:

- data specifying a computer file operable to trigger said execution process;
- data specifying a communication channel operable to trigger said execution

process; and

data specifying an operating system command operable to trigger said execution process.

83. Apparatus as claimed in claim 76, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process.

84. Apparatus as claimed in claim 83, wherein said result data includes data specifying existing configuration data of said destination computer.

85. Apparatus as claimed in claim 84, wherein said execution process is operable to map existing configuration data of said destination computer stored within a configuration data store of said destination computer to said result data to be returned to said source computer.

86. Apparatus as claimed in claim 85, wherein said configuration data store is one of:

- a Windows Registry entry;
- an INI file;
- a DAPI store; and
- a database entry.

87. Apparatus as claimed in claim 85, wherein said result data is passed from said destination computer to said source computer as XML data.

88. Apparatus as claimed in claim 76, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

5

89. Apparatus as claimed in claim 76, wherein an operation that may be performed by said destination computer includes installing a new execution process.

90. Apparatus as claimed in claim 76, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

10

10091415.030702